PCT

WORLD INTELLECTUAL PROPERTY ORGANIZATION
International Bureau

INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification 6 : H04L 9/00	A2	(11) International Publication Number: WO 99/27676 (43) International Publication Date: 3 June 1999 (03.06.99)

(21) International Application Number: PCT/IB98/01834

(22) International Filing Date: 4 November 1998 (04.11.98)

(30) Priority Data:
122106 4 November 1997 (04.11.97) IL

(71) Applicant (*for all designated States except US*): ENCO-TONE LTD. [IL/IL]; P.O. Box 45094, 91450 Jerusalem (IL).

(72) Inventor; and
(75) Inventor/Applicant (*for US only*): LABATON, Isaac, J. [IL/IL]; P.O. Box 45094, 91450 Jerusalem (IL).

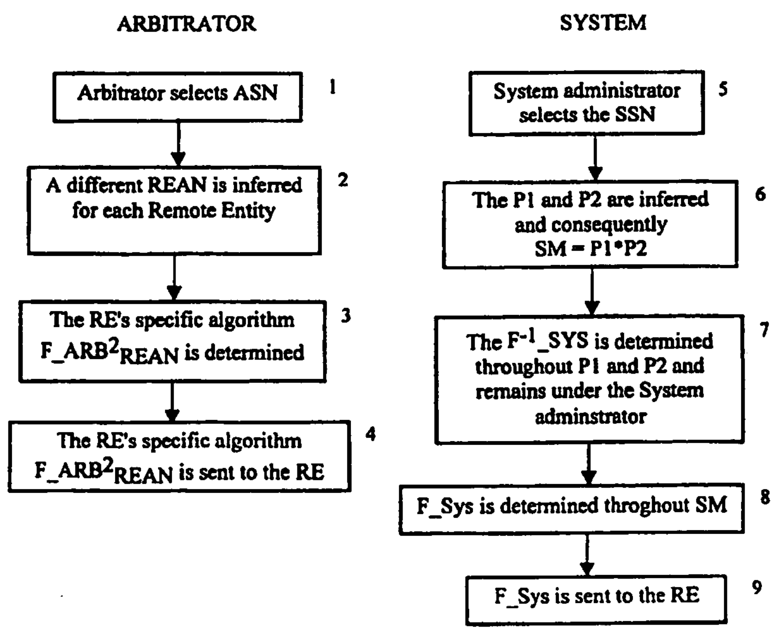
(81) Designated States: AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, CA, CH, CN, CU, CZ, DE, DK, EE, ES, FI, GB, GE, GH, GM, HR, HU, ID, IL, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MD, MG, MK, MN, MW, MX, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TR, TT, UA, UG, US, UZ, VN, YU, ZW, ARIPO patent (GH, GM, KE, LS, MW, SD, SZ, UG, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GW, ML, MR, NE, SN, TD, TG).

Published
Without international search report and to be republished upon receipt of that report.

(54) Title: METHODS AND APPARATUS FOR THE SECURE IDENTIFICATION AND VALIDATION OF THINGS AND EVENTS

(57) Abstract

Methods for non-repudiable, non-trackable, possibly one-way identification and validation of remote entities to identification devices, wherein the identification devices do not require access to databases of remote entity information. An arbitrator entity preferably characterizes and distributes a specific algorithm to each remote entity. An identification device (or system operating an identification device) preferably distributes one reversible algorithm to each remote entity. Each time a remote entity identifies itself to an identification device, it applies its arbitrator provided algorithm to either a time-based variable (one-way identification) or to a challenge provided by the identification device, computing a first result. The remote entity then applies the reversible algorithm to the challenge/time-based variable, to its identification data and to the first computed result, computing a second result which is transmitted to an identification device. The identification device then may apply the reverse algorithm to the second result, computing a presumed challenge/time-based variable, presumed identification data and presumed first result. The identification device then may compare the challenge/time-based variable to the presumed challenge/time-based variable. If they match (within some tolerance for a time-based variable), the identification device transmits the presumed first result, the presumed identification data and the challenge to the arbitrator. The arbitrator then may apply the particular algorithm distributed to that remote entity and apply it to the challenge/time-based variable, thereby computing a valid first result. The arbitrator then may compare the valid first result to the presumed first result. If they match (within a tolerance for time-based variables), the arbitrator may corroborate the authenticity of the identification to the identification device.



FOR THE PURPOSES OF INFORMATION ONLY

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

AL	Albania
AM	Armenia
AT	Austria
AU	Australia
AZ	Azerbaijan
BA	Bosnia and Herzegovina
BB	Barbados
BE	Belgium
BF	Burkina Faso
BG	Bulgaria
BJ	Benin
BR	Brazil
BY	Belarus
CA	Canada
CF	Central African Republic
CG	Congo
CH	Switzerland
CI	Côte d'Ivoire
CM	Cameroon
CN	China
CU	Cuba
CZ	Czech Republic
DE	Germany
DK	Denmark
EE	Estonia
ES	Spain
FI	Finland
FR	France
GA	Gabon
GB	United Kingdom
GE	Georgia
GH	Ghana
GN	Guinea
GR	Greece
HU	Hungary
IE	Ireland
IL	Israel
IS	Iceland
IT	Italy
JP	Japan
KE	Kenya
KG	Kyrgyzstan
KP	Democratic People's Republic of Korea
KR	Republic of Korea
KZ	Kazakstan
LC	Saint Lucia
LI	Liechtenstein
LK	Sri Lanka
LR	Liberia
LS	Lesotho
LT	Lithuania
LU	Luxembourg
LV	Latvia
MC	Monaco
MD	Republic of Moldova
MG	Madagascar
MK	The former Yugoslav Republic of Macedonia
ML	Mali
MN	Mongolia
MR	Mauritania
MW	Malawi
MX	Mexico
NE	Niger
NL	Netherlands
NO	Norway
NZ	New Zealand
PL	Poland
PT	Portugal
RO	Romania
RU	Russian Federation
SD	Sudan
SE	Sweden
SG	Singapore
SI	Slovenia
SK	Slovakia
SN	Senegal
SZ	Swaziland
TD	Chad
TG	Togo
TJ	Tajikistan
TM	Turkmenistan
TR	Turkey
TT	Trinidad and Tobago
UA	Ukraine
UG	Uganda
US	United States of America
UZ	Uzbekistan
VN	Viet Nam
YU	Yugoslavia
ZW	Zimbabwe

METHODS AND APPARATUS FOR THE SECURE IDENTIFICATION AND VALIDATION OF THINGS AND EVENTS

TECHNICAL FIELD

The present invention generally relates to methods for identification, including remote identification and validation of messages. The methods of the present invention provide security in various transactions by validating and identifying various entities, while avoiding the necessity of conventional authentication procedures or the need to store databases of information available to an identification device.

BACKGROUND OF THE INVENTION

Various service providers (e.g., Social Securities, Telecoms (PATS), financial institutions, brokers, banks, merchants, etc.) are often involved in transactions requiring the identification and validation of a remote entity (e.g., an individual, organization, smart card, message, account, etc.). These service providers often provide their services to remote entities over various telecommunications media, for example, Internet, phone lines, etc. Naturally, it is important for these providers to ensure during each transaction that the remote entity is not an imposter. Accordingly, they often employ various identification devices to identify and validate remote entities, these devices being referred to herein as Identification Devices. For ease of discussion, a remote entity authorized to engage in transactions, but perhaps not yet identified and/or authenticated by an Identification Device for a particular transaction, is referred to herein as an "Authorized Remote Entity" or "Authorized Entity."

One method commonly known in the art and employed by Identification Devices for securely identifying a remote entity is to add "authentication" to an otherwise normal identification process. Authentication is typically accomplished by providing an additional piece of information to an Identification Device, e.g., a secret code, along with identification information. This additional information then may be used to corroborate that the identification is accurate and that the remote entity is not an imposter attempting to impersonate an authorized entity. The additional piece of information is often a secret code or a password (e.g., PIN), but

also may be a Dynamic Code, preferably computed using a software implemented algorithm. Alternatively, the additional information may be provided by a token (e.g., Bio-Token) carried by the entity (e.g., individual) to be identified.

Non-variable (i.e., constant or static) information or data (e.g., PIN) can only add limited security to the identification process because a static piece of information eventually may become known to a third party (e.g., potential attacker/impostor/eavesdropper) in which case an authorized entity can easily be impersonated. On the other hand, authentication by means of a variable piece of information (referred to herein as a Dynamic or One Time Code) provides enhanced security.

Currently known methods of authentication which use a Dynamic or One Time Code typically require a prior step of identifying the remote entity to the Identification Device, e.g., by providing a name (e.g., a login name), a serial number, an additional fix code, etc. as part of a message transmitted from a Remote Entity to an Identification Device. This constant part of a message will be referred to herein as an Identification Message. Thus, a method commonly employed by an Identification Device to securely identify a Remote Entity by authentication typically comprises the three following steps:

1) Identification: identify who the Remote Entity is supposed to be, by receiving a constant (non-variable, or at least non-constantly-variable) piece of information, referred to herein as an Identification Message;

2) Database Search: the Identification Devices searches a database containing the Authorized Entity's secret information or computing keys, to compute a dynamic piece of information (referred to herein as a Valid Dynamic Code) which is associated with and expected from the Authorized Entity at that particular moment; and

3) Authentication: the Identification Device compares the Valid Dynamic Code (computed at the Identification Device) with a Dynamic Code received from the Remote Entity (referred to as the Received Dynamic Code) to check if both codes match; if so, the Identification Device corroborates the identification of the Remote Entity as being the Presumed Entity.

A variation of the above-described authentication method is referred to as the Challenge and Response method, comprising the following steps:

1) The Remote Entity is identified (as described in step 1 above);

2) The Remote Entity receives a Challenge generated and sent by the Identification Device and computes a Response, the Response playing the role of a Dynamic Code;

3) The Identification Device, after identifying the Remote Entity (the Pre-Authentication Identification), requires a database to determine the expected response to the challenge, for that Remote Entity at that moment.

Each of the authentication schemes described above requires the Identification Device to employ a database or look-up table. Naturally, each database must be maintained and updated, which creates problems associated with the management of keys, synchronized database updates, etc. Furthermore, these problems become acute when a service provider utilizing an authentication process has a multitude of Identification Devices disseminated through several countries. Accordingly, methods of authentication are needed which overcome limitations and drawbacks associated with the use of databases in authentication methods currently known in the art.

Another problem associated with conventional schemes for Remote Identification is the possibility of "repudiation" by an identified and authenticated Remote Entity. For example, a Remote Entity, which has been identified and authenticated as being an Authorized Entity, may later deny the genuineness of a particular communication or event under scrutiny. To illustrate, in the case of a Gambling Service Provider (although identification and authentication techniques may apply to any service provider, Gambling Service Providers are used for this example), Remote Entities (e.g., gamblers) may place bets from remote locations and pay for those bets using Credit Cards. Naturally, before a particular Remote Entity places any bets, the Gambling Service Provider identifies and authenticates that Remote Entity by a procedure similar to those described above. Once the bets have been placed, one of the Remote Entities wins a prize, while all of the remaining Remote Entity gamblers lose. This situation presents an opportunity for any number of losing Remote Entities to repudiate their particular betting transaction, including the identification and authentication process, claiming that they never made the transaction/bet, and that the Gambling Service Provider fabricated the transaction or made a mistake. Because each Remote Entity is authenticated by the Provider's Identification Device, and further because the provider includes a database containing secret information, the

Provider has the capability to compute as many Valid Dynamic Codes as the Gambling Service Provider may desire, and an unscrupulous Gambling Service Provider thereby has the ability to fabricate transactions. Accordingly, when a Remote Entity repudiates a transaction, there is no way to prove whether the Gambling Service Provider fabricated the transaction or the Remote Entity has repudiated a valid transaction. Of course, if all the losing Remote Entities repudiate their transactions, the effect on the Gambling Service Provider may be disastrous.

As illustrated in the example above, present methods of authentication intrinsically are subject to the negative effects of transaction repudiation, due to the fact that the receiving/identifying/authenticating side of each transaction has the capability to compute a secret Dynamic Code as accurately as the Remote Entity. Accordingly, new methods are needed which avoid Remote Entity repudiation of transactions.

A further drawback of authentication methods known in the art and described above is the fact that a Remote Entity is trackable. In other words, an eavesdropper may follow every transaction made a particular Remote Entity because that Remote Entity transmits the same constant identification information for every transaction. This ability to track a Remote Entity creates a lack of security and privacy for many Remote Entities (e.g., especially government officers, ministers, police officers, etc.). Accordingly, new methods of identification are needed which avoid the trackability of Remote Entity transactions.

BRIEF SUMMARY OF THE INVENTION

The present invention discloses methods for secure identification of entities, whereby no authentication process is necessary. In other words, the present invention comprises various One Step/One Way Identification Procedures whereby only a secure identification step is required, without the need of an additional authentication step. Furthermore, these methods of identification help to avoid the possibility of impersonation because a part of each identification process is variable and valid only one time. Consequently, the identification process cannot be intercepted and re-used, is non-repudiable and is non-trackable. In addition, by utilizing the methods of the present invention, the need for an Identification Device, as described above, to have access to a database of authentication data is alleviated.

The invention comprises the use of Reversible Algorithms, as shown below, in accordance with the principles disclosed in US patent 5,524,072 issued to Labaton et al., hereby incorporated by reference.

According to a preferred method of the present invention, reversible algorithms (e.g., mathematical algorithms) may be used in conjunction with a Challenge-Response method similar to that described above, wherein a Identification Device generates a random challenge. This challenge then may be transmitted to a Remote Entity. The Remote Entity generates a substantially or totally dynamic response (i.e., little or no constant part), and such response constitutes "secure identification information." It is important to emphasize at this point that, according to a preferred embodiment of this invention, an Identification Device does not need to know the identity of any Presumed Entity in order to identify a particular Remote Entity. Accordingly, an Identification Device is capable of checking the identification of a Remote Entity without the need of a database and without the need to know in advance who the Remote Entity is supposed to be (i.e., without the need to know an Authorized Entity).

This invention further provides methods for non-repudiable identification, which means that the Remote Entity will not be able to claim that a Service Provider fabricated the identification message. This is accomplished by providing three different entities, namely, a Remote Entity, a Service Provider/Identification Device and an Arbitrator, wherein no single entity has access to all of the necessary data.

The present invention further provides for systems and methods wherein a Response computation is composed of more than one step, the first step being a computation of the result (R1) inferred from the Arbitrator Seed number. The result R1 will be one of the arguments of the Reversible Algorithm.

In accordance with a further aspect of the present invention, an anti-repudiation feature comprises the following steps: first, a Service Provider receives an identification message from a Remote Entity; second, the Service Provider applies the reverse of the Reversible Algorithm to the signature, and recuperates the original arguments, including R1. Because R1 is computed by the remote entity using the Arbitrator seed number, which is not known to the Service Provider, that means that a correct R1, validated and corroborated by the Arbitrator, can come only from the Remote Entity, and can not be fabricated by the service provider.

BRIEF DESCRIPTION OF THE DRAWING FIGURES

The subject of the invention will, hereinafter, be described in conjunction with the appended drawing figures, wherein like numerals designate like elements, and:

Figure 1 is a block flow diagram of an exemplary initialization process for an Arbitrator and a System (e.g., a Service Provider which operates an Identification Device, ID Node, or multitude ID Nodes) according to the methods of the present invention;

Figure 2 is a block flow diagram of an exemplary identification process for a challenge-response type identification according to the methods of the present invention;

Figure 3 is a block flow diagram of an exemplary anti-repudiation process;

Figure 4 is a block flow diagram of an exemplary identification process according to the methods of the present invention;

Figure 5 is a block flow diagram of an exemplary time-based identification process, with drift correction, according to the methods of the present invention;

Figure 6 is a block flow diagram of a further exemplary time-based identification process, with drift correction, and wherein an Identification Device has access to a database; and

Figure 7 is a graphical representation of a preferred FTolerance function.

DETAILED DESCRIPTION

The present invention provides methods for secure identification of a Remote Entity by an Identification Device, wherein the Identification Device may not require the use of a dedicated local database. Furthermore, the methods of the present invention may provide for one-way, non-repudiable, non-trackable identification. Typically, an Identification Device is operated by a Service Provider, or any other entity requiring secure identification of Remote Entities, to engage in transactions with those entities (for simplicity, referred to herein as a Service Provider).

In accordance with a first method of the present invention, a reversible algorithm (referred to herein as a Reversible Function) F_SYS_{SSn} may be selected by a Service Provider out of a family of possible algorithms, preferably by selecting a System Seed Number (or SSN).

The Service Provider then may distribute the reversible algorithm to any or all of the Remote Entities authorized to engage in transactions with the Service Provider (Authorized Remote Entities). The Service Provider also may provide a specific Remote Entity Identification number (referred to herein as C_ID) to each of the Authorized Remote Entities (e.g., a driver's license number).

Each time a Remote Entity is to be identified by a Service Provider (e.g., for a particular transaction between the Service Provider and the Remote Entity), the Service Provider may select a random number (referred to herein as Ch or Challenge) and transmits it to the Remote Entity. The Remote Entity then may apply a function F_SYS_{ssn} to the parameters C_ID and Ch, thereby generating response R2 as follows:

$$F\_SYS_{ssn}\,(C\_ID, Ch) = R2.$$

The response (R2) then may be transmitted to the Service Provider's Identification Device. At the Identification Device, the reverse of the system function F_SYS_{ssn} (referred to herein as $F^{-1}\_SYS_{ssn}$) may be applied to R2, thereby recuperating C_ID and a number presumed to be Ch, referred to herein as Ch':

$$F^{-1}\_SYS_{ssn}\,(R2) = C\_ID, Ch'.$$

The Identification Device then may compare Ch (sent) to Ch' (received/computed) and determine whether they match. If they match, C_ID is a correct and valid identification.

In a preferred exemplary implementation of the above-described method, the present invention may utilize a Rabin Algorithm (known in the art). Accordingly, the Service Provider may select two large prime numbers, P_1 and P_2, each of them being congruent with 7(mod 8), or alternatively, select a System Seed Number (SSN), from which two such prime numbers may be consequently inferred (See Figure 1, blocks 5 and 6). These two prime numbers will determine, as parameters, the Reverse Algorithm, as explained below, but the prime numbers will not be transmitted or communicated to the Remote Entity, and will remain as secret system keys under the supervision of the Service Provider (See Figure1, block 7). The Service Provider then may calculate the product $P_1 * P_2 = SM$ (referred to herein as a System Module) and send the System Module (SM) to Remote Entities, not necessarily openly, but preferably embedded in a function F_SYS_{ssn} (See Figure 1, block 8). The F_SYS_{ssn} may be masked on a chip.

A preferred exemplary model for the function F_SYS_{ssn} may be specified as a function $FF_M(x)$ as follows:

$$FF_M(x) = x^2 (\text{mod } M).$$

The Remote Entity then may make the following computations:

$FF_{SM}(Ch) = L_1$ and then

$FF_{SM}(L_1) = L_2$ and then

$FF_{SM}(L_2) = L_3$ and then

$FF_{SM}(L_3) = L_4$ and then

..............=...............

and so on until q (where q is any natural number),

$FF_{SM}(L_{q-1}) = L_q$.

And then, defining the function LSB(X) as the 'Least Significant (n)Bit(s)' of X (where n is any natural number), the Remote Entity may make the following computations:

$FF_{SM}(Lq) = T1$ and then $LSB(T1) = c1$

$FF_{SM}(T1) = T2$ and then $LSB(T2) = c2$

$FF_{SM}(T2) = T3$ and then $LSB(T3) = c3$

$FF_{SM}(T3) = T4$ and then $LSB(T4) = c4$

..............=.........................= ...

and so on until n, where n is the amount of digits of the concatenation $Ch \circ C\_ID$,

$FF_{SM}(Tn\text{-}1) = Tn$ and then $LSB(Tn) = cn$.

The Remote Entity then may concatenate c1,...,cn and refer to the concatenation as C as follows:

$$C = c1 \circ c2 \circ c3 \circ c4 \circ \circ \circ \circ \circ \circ \circ \circ \circ cn$$

(wherein $\circ$ stands for concatenation).

Then, the Remote Entity may bitwise-xor C with the concatenation $Ch \circ C\_ID$ as follows (where "xor" corresponds to the "exclusive or" function):

$$(c1 \circ c2 \circ c3 \circ c4 \circ \circ \circ \circ \circ \circ \circ \circ \circ cn) \otimes (Ch \circ C\_ID) = V$$

(wherein $\otimes$ stands for bitwise-xor).

Then, the Remote Entity may compute Tn+m (m being any natural number) in the following manner:

FF_{SM} (Tn) = Tn+1

.... = ...

and so on until n+m,

FF_{SM} (Tn+m-1) = Tn+m.

The Remote Entity then may concatenate V and Tn+m as follows, resulting in R2 (referred to herein as a Cipher):

$R2 = V \circ Tn+m$

At this point, a Remote Entity may transmit the Cipher R2 to an Identification Device. Because the Identification Device (or, e.g., Service Provider operating the Identification Device) knows the prime numbers P_1 and P_2, it may reverse the algorithm and recover Ch and C_ID, as follows:

The Identification Device may use the Euclidean Algorithm in conjunction with P_1 and P_2, to compute Ll and L2, such that:

$L_1 P_1 + L_2 P_2 = 1(\text{mod SM}).$

K1 and K2 may be defined as follows:

$K_1 = L_1 P_1$ and $K_2 = L_2 P_2$.

Then, the following computations may be made as follows:

$Y_1 = Tn+m \ (\text{mod } P_1)$ and $Y_2 = Tn+m \ (\text{mod } P_2)$.

Then, the following computations may be made as follows:

$Z_1 = Y_1^{((P_1+1)/4)} \ (\text{mod } P_1)$ and,

if $Z_1^{((P_1-1)/2)} = -1 \ (\text{mod } P_1)$ then $Z_1 = P_1 - Z_1$.

Then, the following computations may be made as follows:

$Z_2 = Y_2^{((P_2+1)/4)} \ (\text{mod } P_2)$ and,

if $Z_2^{((P_2-1)/2)} = -1 \ (\text{mod } P_2)$ then $Z_2 = P_2 - Z_2$.

Then, the following computations may be made as follows:

$Tn+m-1 = Z_1 K_2 + Z_2 K_1$.

Then, applying the same procedure, compute Tn+m-2, and so on, obtaining all of the T_i until T_0.

The Identification Device then may proceed as follows:

Taking Ti, the following computations may be made as follows:

C_i = LSB(Ti) for I = 1,,n.

The results then may be concatenated as follows:

$C = c1 \circ c2 \circ c3 \circ c4 \circ \circ \circ \circ \circ \circ \circ \circ \circ cn.$

Then, the Identification Device may XOR C with V as follows:

[$V = (c1 \circ c2 \circ c3 \circ c4 \circ \circ \circ \circ \circ \circ \circ \circ \circ cn) \otimes (Ch \circ C\_ID)$, sent from the Remote Entity],

and then compute as follows:

$(c1 \circ c2 \circ c3 \circ c4 \circ \circ \circ cn) \otimes V =$

$(c1 \circ c2 \circ c3 \circ c4 \circ \circ \circ cn) \otimes (c1 \circ c2 \circ c3 \circ c4 \circ \circ \circ cn) \otimes (Ch \circ C\_ID) = (Ch \circ C\_ID)$

Having recuperated the C_ID, the identification is complete. Recuperation of the Ch then may be used to authenticate the identification, to the extent that Ch should be exactly the same as the Ch sent to the Remote Entity.

It is worth noting that the example shown represents an extremely secure methodology for identifying a Remote Entity because the sensitive secrets, namely, the knowledge regarding how to factorize the SSN (namely P_1 and P2) is only available at the Identification Device and not available to the Remote Entity (e.g., not stored in a token carried by the Remote Entity). Of course, any suitable mathematical (or other) manipulations or operations may be employed in the context of the present invention.

The present invention further provides methods for identifying one or more Remote Entities to a Central Identification Device (e.g., operated by a Service Provider) wherein the identification process is arbitrated by an Independent Arbitrator Entity. These methods preferably comprise the following steps:

a) The Independent Arbitrator Entity may characterize a specific algorithm (or a plurality of algorithms) for each of the Remote Entities, and then distribute each algorithm to its respective Remote Entity;

b) The Central Identification Device (e.g., operated by a Service Provider) may distribute one (or more) reversible algorithm to each of the Remote Entities;

c) Each time a Remote Entity is to be identified, the Remote Entity may apply the Independent Arbitrator Entity's algorithm to a challenge (Ch) provided by the Central Identification Device, thereby providing a first result R1. The Remote Entity then may apply the Central Identification Device's reversible algorithm to the challenge Ch, to its identification data C_ID, and to the said first result R1, thereby computing a second result R2 (the Cipher). The Remote Entity then may transmit the second result R2 to the Central Identification Device. The Central Identification Device then may apply the reverse of the reversible algorithm to the received second result R2, thereby computing a presumed challenge Ch', a C_ID, and a presumed first result R1. The Identification Device then may compare the transmitted challenge Ch to the computed presumed challenge Ch'. If they are identical, the Central Identification Device may transmit the presumed first result R1 to an Independent Arbitrator Entity together with the Remote Entity's identification data (which may be C_ID or other data, for example a Remote Entity's Serial Number) and the challenge Ch. The Arbitrator Entity then may retrieve the specific algorithm transmitted from the specific Remote entity and apply the algorithm to the challenge Ch, thereby computing a true first result R1. The Arbitrator Entity then may compare the true first result R1 with the presumed first result R1 and, if they are identical, the arbitrator entity may corroborate the identification to the Central Identification Device. Once the Central Identification Device has received the corroboration from the Arbitrator Entity, it may accept the presumed identification data C_ID as being true identification data corresponding to the Remote Entity.

These methods (or other suitable operations) utilizing three entities (Remote Entity, Central Identification Device and Arbitrator Entity), provide for non-repudiable and non-trackable secure identifications methodology. An exemplary implementation of these methods may comprises the following steps:

a) In addition to Remote Entities and Identification Devices, utilize a third independent party, referred to herein, for example, as an Arbitrator Entity;

b) The Arbitrator Entity may select a first algorithm (referred to herein as $F\_ARB^1{}_{ASN}$) which preferably is selected from a family of algorithms F_ARB^1, preferably by the means of a seed number (referred to herein as an Arbitrator Seed Number = ASN) (See Figure 1 Block 1).

An important purpose of this algorithm ($F\_ARB^1{}_{ASN}$) is to generate a different pseudo-random number for each Remote Entity (See Figure 1, block 2). This pseudo-random number, referred to herein as REAN (Remote Entity's Arbitrator Number), determines the algorithm ($F\_ARB^2{}_{REAN}$) (see Figure 1, blocks 3 and 4) which may correlate the transaction related number Ch and, optionally C_ID, with an arbitrator inferred number R1. The REAN is preferably the result of the product of two large prime numbers, p'1 and p'2, both of which may be congruent with 3(mod4). Hence the computed R1 may be different and specific for each transaction.

Accordingly, the algorithm $F\_ARB^1{}_{ASN}$ is suitably applied to a Remote Entity's openly known number, for example, the Remote Entity's Serial Number or any other publicly known number associated with the Remote Entity, to generate the REAN (Remote Entity's Arbitrator Number), as follows:

$F\_ARB^1{}_{ASN}$ (Serial Number) = REAN.

The second algorithm selected by the Arbitrator is referred to herein as $F\_ARB^2{}_{REAN}$, which also preferably is selected from a family of algorithms by means of the previously obtained Remote Entity Arbitrator Number REAN (See Figure 1, block 3).

It is worth noting that an Arbitrator only needs to select one number, namely the ASN, which is suitably the same for all the cases belonging to such Arbitrator. In contrast, the REAN is advantageously different and preferably dedicated for each Remote Entity, and the R1 is different for each transaction and inferred from the ASN to further enhance security.

The Remote Entity may then apply the $F\_ARB^2{}_{REAN}$ algorithm to Ch (and, optionally, to C_ID), thereby getting a response R1 (see Figure 2, block 11), as follows:

$F\_ARB^2{}_{REAN}$ (C_ID, Ch) = R1.

At this point, a Remote Entity may apply the F_SYS_{ssn} algorithm to Ch, C_ID and R1, thereby retrieving the Cipher R2 (see Figure 2, block 12), as follows:

F_SYS_{ssn} (Ch , C_ID, R1) = R2 (the Cipher).

The Remote Entity may then transmit Cipher R2 to an Identification Device (e.g., ID Node or System Administrator) (See Figure 2, block 13).

The Identification Device may then receive Cipher R2 (see Figure 2, block 14). The Identification Device, which knows the values of P_1 and P2, then may non-repudiably identify the Remote Entity, without using any database, by applying the reverse of F_SYS_{ssn} to R2. Accordingly, $F^{-1}\_SYS_{ssn}$ is applied to R2, thereby retrieving the original arguments, namely, C_ID, Ch' and R1, as follows:

$$F^{-1}\_SYS_{ssn}(R2) = C\_ID, Ch', R1.$$

If Ch' is sufficiently identical to the Ch transmitted, then the C_ID is validated as being authentic, e.g., an authentic Entity Identification Number (See Figure 2, blocks 15 and 16).

R1 then may be preserved to refute repudiations of such transactions. That is, cases where the Remote Entity denies that he/she/it had a Cipher (e.g., R2), or, in other words, when the Remote Entity claims that a Cipher (e.g., R2) was fabricated by the Identification Device and/or the controller of such a device (See Figure 2, block 17).

Referring now to Figure 3, in the case of repudiation problems or, alternatively, for each desired case, the Identification Device may provide the Arbitrator with a Remote Entity's Serial Number, the Ch, the Presumed R1, and, optionally, the C_ID (see Figure 3, blocks 19,20 and 21).

The Arbitrator may then apply the $F\_ARB^1{}_{ASN}$ algorithm to the Serial Number, thereby obtaining the REAN, as follows:

$F\_ARB^1{}_{ASN}$ (Serial Number) = REAN (See Figure 3, block 22).

The Arbitrator may then apply the $F\_ARB^2{}_{REAN}$ algorithm to Ch and, optionally, to C_ID, and thereby obtain the true R1 (see Figure 3, block 23) as follows:

$F\_ARB^2{}_{REAN}(C\_ID, Ch) = R1$, which is the true R1.

The arbitrator may then determine whether the true R1 is identical to the Presumed R1 (see Figure 3, blocks 24 and 25). If R1 and the Presumed R1 are not identical, then R2 may be a fabrication (See Figure3, block 27). On the other hand, if R1 matches the presumed R1, it may be concluded that the R2 is coming from an Authorized Remote Entity, because the Identification Device has no means with which to compute (fabricate) a true R1 (see Figure 3, block 26).

According to another preferred exemplary implementation of the methods of the present invention, the Remote Entity may utilize an algorithm $F\_ARB^2_{REAN}$, characterized by an Arbitrator by means of a REAN, to compute the transaction specific R1, using the transaction specific Challenge Ch transmitted by the Identification Device (e.g., ID Node/ System Administrator), as follows:

the Remote Entity may compute:

$$M_1 = Ch^2 (\mathrm{mod\ REAN})$$

$$M_2 = M^2_1 (\mathrm{mod\ REAN})$$

.....=.....................

and so on, until p (any natural number)

$$S_1 = M_p = M^2_{p-1} (\mathrm{mod\ REAN}).$$

The Remote Entity then may compute:

$$C'_1 = LSB (S_1).$$

Accordingly,

$$S_2 = S^2_1 (\mathrm{mod\ REAN}) \quad \rightarrow C'_2 = LSB (S_2)$$

and so on, until n:

$$S_n = S^2_{n-1} (\mathrm{mod\ REAN}) \quad \rightarrow C'_n = LSB (S_n).$$

The result is the following concatenation:

$$R1 = C'_1 \circ \circ \circ \circ \circ \circ \circ \circ \circ C'_n.$$

Accordingly, the above-described step $F\_ARB^2_{REAN} (Ch) = R1$ thereby may be accomplished.

According to a further preferred implementation of the methods of the present invention, a preferred implementation of the function F_SYS_{ssn} is now presented. In accordance with this aspect of the present invention, a System Seed Number (SSN), selected by an administrator, may be used to infer a SM (System Module, as discussed above). The Remote Entity may make the following computations:

$$T_1 = R1^2 (\mathrm{mod\ SM}) \Rightarrow C_1 = LSB (T_1)$$

$$T_2 = T^2_1 (\mathrm{mod\ SM}) \Rightarrow C2 = LSB (T_2)$$

...... = ... ⇒......=..........................

and so on, to:

$$T_n = T^2{}_{n-1} \text{ (mod SM)} \Rightarrow C_n = \text{LSB } (T_n)$$

$$T_{n+1} = T^2{}_n \text{ (mod SM)}$$

.... =................

and so on, to:

$$T_{n+m} = T^2{}_{n+m} \text{ (mod SM)}.$$

The C's then may be concatenated as follows:

$$C = C_1 \circ \circ \circ \circ \circ \circ \circ \circ \circ C_n.$$

A Cipher (R2) message then may be computed as follows:

$$R2 = [(C\_ID \circ CH)] \otimes C] \circ T_{n+m}.$$

Accordingly, the above-mentioned step F_SYS_{ssn} (Ch) = R2 thereby may be accomplished.

Obviously, while a C_ID is always the same for a particular Remote ~~Entity~~, the Cipher should be different from any prior Cipher (R2), to the extent that the Ch is different from any prior Ch transmitted to a particular Remote Entity.

The Remote Entity may transmit the Cipher R2 to an Identification Device (e.g., ID Node/System Administrator) and, because the Identification Device knows the prime numbers P_1 and P2, the Identification Device may reverse the algorithm and recover Ch and C_ID.

The Identification Device (e.g., ID Node/ System Administrator) may use the Euclidean Algorithm (or other suitable Algorithms) in conjunction with P_1 and P_2 to compute what is referred to as the reverse algorithm or Reverse Algorithm, $F^{-1}\_SYS_{ssn}$. A preferred exemplary implementation of this computation is as follows:

a) compute L1 and L2 such that $L_1 P_1 + L_2 P_2 = 1 \text{(mod SM)}$;

b) define $K_1 = L_1 P_1$ and $K_2 = L_2 P_2$;

c) compute:

$Y_1 = Tn{+}m \text{ (mod } P_1)$ and

$Y_2 = Tn{+}m \text{ (mod } P_2)$;

d) compute:

$Z_1 = Y_1^{((P_1+1)/4)} \pmod{P_1}$

and, if $Z_1^{((P_1-1)/2)} = -1 \pmod{P_1}$ then $Z_1 = P_1 - Z_1$; and

$Z_2 = Y_2^{((P_2+1)/4)} \pmod{P_2}$

if $Z_2^{((P_2-1)/2)} = -1 \pmod{P_2}$ then $Z_2 = P_2 - Z_2$;

e) compute:

$T_{n+m-1} = Z_1 K_2 + Z_2 K_1$;

and, applying the same procedure, compute:

T_{n+m-2}

and so on, obtaining all the Ti until T_0 is recuperated.

The Identification Device (e.g., ID Node/System Administrator) then may proceed as follows:

a) using Ti, compute: ci = LSB(Ti)

and so on for i = n,n-1,...,1 (T_1 is also recaptured);

clearly, the Reverse algorithm applied to T_1 generates the R1, which is stored together with the correspondent Ch for potential repudiation cases;

b) the ci results are concatenated as follows:

$C = c1 \circ c2 \circ c3 \circ c4 \circ \circ \circ \circ \circ \circ \circ \circ \circ cn$

c) the Identification Device (e.g., ID Node/System Administrator) then may XOR C with V, thereby obtaining:

$(c1 \circ c2 \circ c3 \circ c4 \circ \circ \circ cn) \otimes V =$

$(c1 \circ c2 \circ c3 \circ c4 \circ \circ \circ cn) \otimes (c1 \circ c2 \circ c3 \circ c4 \circ \circ \circ cn) \otimes (Ch \circ C\_ID) = (Ch \circ C\_ID)$.

Having recuperated the C_ID, the identification may be complete. The recuperation of Ch then may be used to authenticate the identification, to the extent that Ch should be exactly the same as the Ch sent to the Remote Entity.

If a Remote Entity later repudiates the transaction, the Identification Device (e.g., ID Node/System Administrator) may provide the Arbitrator with the Remote Entity's Serial

Number, the transaction Ch, the presumed R1 and, optionally, the C_ID. The Arbitrator then may apply the $F\_ARB^1{}_{ASN}$ to the Serial number, thereby obtaining the REAN as follows:

$$F\_ARB^1{}_{ASN} \text{ (Serial Number)} = \text{REAN}.$$

The Arbitrator then may apply the $F\_ARB^2{}_{REAN}$ algorithm to Ch and, optionally, to C_ID, thereby obtaining the true R1, as follows:

a) the Arbitrator may compute:

$$M_1 = Ch^2 \pmod{REAN}$$

$$M_2 = M^2{}_1 \pmod{REAN}$$

and so on, until

$$S_1 = M_{40} = M^2{}_{39} \pmod{REAN},$$

(wherein 39 and 40 are examples only and may differ);

b) the Arbitrator may compute:

$$C'_1 = LSB\ (S_1);$$

c) consequently:

$$S_2 = S^2{}_1 \pmod{REAN} \qquad \rightarrow C'_2 = LSB\ (S_2)$$

and so on, up to

$$S_n = S^2{}_{n-1} \pmod{REAN} \qquad \rightarrow C'_n = LSB\ (S_n).$$

Accordingly, the result is the true R1 for this specific Remote Entity and for this specific transaction (Ch), wherein:

$$R1 = C'_1 \circ \circ \circ \circ \circ \circ \circ \circ \circ C'_n.$$

The Arbitrator may then check whether the true R1 is identical to the Presumed R1 claimed by the Identification Device (e.g., ID Node/system administrator). If they do not match, the R2 is a fabrication. On the other hand, if the two numbers match, it is confirmed that the R2 is from the Remote Entity because the Identification Device has no means with which to compute (fabricate) a true R1.

It is worth noting that the example shown represents an remarkably secure methodology to identify a Remote Entity. The sensitive secrets, e.g., the knowledge on how to factorize the

SSN, is available only to the Identification Device (e.g., ID Node/ System Administrator) and is not available to the Remote Entities (e.g., not in their tokens).

According to another preferred exemplary implementation of the methods of the present invention, the methods and algorithms described above may be used, but, instead of having the Identification Device send a challenge (Ch) to the Remote Entity, the Remote Entity may use the Cipher's Generation Time. The time of the moment of computing the Cipher, the Generation Time and Date, the GMT and Date, or some other suitable time-based variable having a particular resolution, e.g., of seconds. For simplicity, this type of time-based variable will be referred to herein as Generation Time, or Gtime (see Figure 4, block 10b).

Referring to Figure 4, a Remote Entity may proceed as in Figure 4, illustrating a non-repudiable, non-trackable, one-way identification method. The Remote Entity may transmit a message, which identifies the Remote Entity, to the Identification Device (e.g., ID Node, or one of various ID Nodes of a particular Service Provider), and this ends the protocol. Accordingly, during the time that the identification is being made on-line, both sides, Remote Entity and Identification Device, know the corresponding Gtime/Reception Time (see Figure 4, block 14b) and, consequently, the Identification Device may verify the identification (See Figure 4, blocks 15b and 16b). If the Identification is not on-line, the Generation Time used in the computation may be communicated openly to the Identification Device.

With regard to on-line transactions, to overcome possible drift between the time measured by the Remote Entity and the time measured by the Identification Device, two successive identification procedures may be accomplished during the same identification process (see Figure 5). The time elapsed between the two identification procedures also may be determined by the Identification Device (see Figure 5, block 19d), which will signal/ trigger the Remote Entity to send a new Cipher. Because the time elapsed is set by the Identification Device, this process avoids the possibility of an impostor using two pre-recorded Ciphers in an attempt to defraud the system. Hence, measuring the Absolute Value of the drift difference (see Figure 5, block 27d) the Identification Device can overcome the drift problem without compromising security.

Another preferred exemplary implementation of the methods of the present invention comprises a process similar to that described in Figure 5, but instead of using the Gtime a Remote Entity may use a Remote entity's specific function of the Gtime, referred to herein as

F_T_RE(Time), which preferably is a linear function. The Identification Device (e.g., ID Node/ System Administrator) will require the knowledge to compute F_T_RE(Time), and therefore, a database may be necessary.

As an example, a function of Gtime may be utilized wherein F_T_RE(Time) = RE_Time = Time + RE_cte, and where RE_cte is a constant value, and is different for each Remote Entity (see Figure 6). Because, according to this particular implementation, the Identification Device may use a database anyway, it may be convenient to preserve the RE_Time and RTime of the previous transaction (referred to herein as RE_Time_pre and RTime_pre) in a database available to the Identification Device. This is in order to update the drift while the RE_cte value is being updated, which absorbs the drift as is shown in Figure 6.

The Identification Device, after receiving the RE_Time embedded in the Cipher and registering the RTime (Reception Time), will compare the difference:

RE_Time - RTime

with the previous difference (last recorded transaction):

RE_Time - RTime

which will be referred to herein as:

RE_Time_pre - RTime_pre.

This difference should be less than a tolerance value. The tolerance value may be a function (FTolerance) of the absolute value of the difference between RTime and Rtime_pre, as follows:

FTolerance(RTime - RTime_pre).

A preferred specification of FTolerance is shown in Figure 7.

Although the invention has been described herein using specific examples of the various methods encompassed by the present invention, variations or alterations of the methods presented herein do not represent a departure from the spirit of the invention as set forth in the specification and claims. For example, the methods of the present invention may include the addition of encryption steps. Furthermore, the addition of DES to the cipher or permutations of it; the addition of transaction data to the cipher, whether such transaction data is encrypted or not; and/or the addition of Error Correction algorithms. Moreover, the sequential order of the steps or sub-steps illustrated herein are used only to explain the methodology presented, and do not limit or define the scope of the invention. Accordingly, slight variations in the

implementation and/or sequential order the method steps described herein do not represent a departure from the spirit and scope of the invention.

Although the invention has been described herein using a general method for identification, many possible implementations of the methods presented by the invention are possible and remain within the scope of the invention. For example, these methods may be implemented in part wherein the Remote Entities are tokens or other portable devices (e.g., smart cards), or are the carriers of such devices. Moreover, the Identification Devices described herein may be in the form of PC's, ATMs, kiosks, or the like. Furthermore, the above-described methods may be masked or otherwise embedded into chips, and would not represent a departure from the spirit of the present invention.

Although the invention has been described herein in conjunction with the appended drawing figures and specific functions, those skilled in the art will appreciate that the scope of the invention is not so limited. Various modifications in the selection and arrangement of the various components, method and steps discussed herein may be made without departing from the spirit of the invention as set forth above.

We claim:

1. An identification method for a multitude of to-be-identified-entities against a central identification entity, the said identification arbitrated by an independent arbitrator entity comprising the steps of:

the independent arbitrator entity characterizing specific algorithms for each, and distributing the said algorithms to each of the to-be- identified entities;

the central identification entity distributing one reversible algorithm for all and to all of the to-be-identified-entities;

and for each identification operation, the to-be-identified-entity:

applies the independent arbitrator entity's algorithm to a challenge selected by the central identification entity for such identification operation computing a first result, and then

applies the central identification entity's reversible algorithm to the said challenge, to its identification data, and to the said first result computing a second result (cipher),

and the to-be-identified-entity transmits the said second result to the central identification entity;

whereas

the central identification entity applies the reverse of the reversible algorithm to the received second result, computing a presumed challenge, a presumed to-be-identified entity's identification data, and a presumed first result;

and whereas the central identification entity compares the challenge previously send to the said presumed challenge, and if they are identical,

the central identification entity sends to the independent arbitrator entity the presumed first result,

together with the to-be-identified presumed entity identification data or other identification data of such entity, and

the said challenge,

and whereas the arbitrator entity retrieves the specific algorithm sent by him to the said to-be-identified entity and applies the said specific algorithm to the challenge computing the true first result, and compares such true first result with the presumed first result, and, if eventually are identical, the arbitrator entity corroborates the veracity of the identification to the central identification entity;

and whereas the central entity, having received the corroboration from the arbitrator entity, accepts the presumed identification data as true identification data corresponding to the to-be-identified-entity.

2. An identification method for a multitude of to-be-identified-entities against a central identification entity, the said identification arbitrated by an independent arbitrator entity, as in claim 1, but instead of receiving a challenge from the central identification entity, the to-be-identified-entity computes the time and date of the moment, applies the independent arbitrator entity's algorithm to the said time and date computing a first result, and then applies the central identification entity's reversible algorithm to the said time and date, to its identification data, and to the said first result computing a second result (cipher), and the to-be-identified-entity send the said second result to the central identification entity;

and whereas the central identification entity:

records the reception time of the cipher sent by the to-be-identified entity

the central identification entity applies the reverse of the reversible algorithm to the received second result, computing a presumed time and date, a presumed to-be-identified entity's identification data, and a presumed first result;

and whereas the central identification entity compares the said reception time to the said presumed time and date, and if they are alike according to a pre-established tolerance, the central identification entity sends to the independent arbitrator entity the presumed first result,

together with the to-be-identified entity identification data or other identification data of such entity, and

the said presumed time and date,

and whereas the arbitrator entity retrieves the specific algorithm sent by him to the said to-be-identified entity and applies the said specific algorithm to the said presumed time and date received from the central identification entity, computing the true first result, and compares such true first result with the presumed first result, and, if eventually are identical, the arbitrator entity corroborates the veracity of the identification to the central identification. entity;

and whereas the central entity, having received the corroboration from the arbitrator entity, accepts the presumed identification data as true identification data corresponding to the to-be-identified-entity.

3. A system for secure identification comprising:

an arbitrator configured to store and provide a different algorithm for each of a plurality of remote entities, comprising a first processor having access to a first memory; and

an identification device configured to provide a reversible algorithm to each of said remote entities, comprising a second processor having access to a second memory;

wherein each of said remote entities comprises a remote entity memory and a remote entity processor, and is configured to store one of said arbitrator algorithms, said reversible algorithm and remote entity identity information; and

wherein said first processor cannot access said second memory and said second processor cannot access said first memory.

FIGURE 1

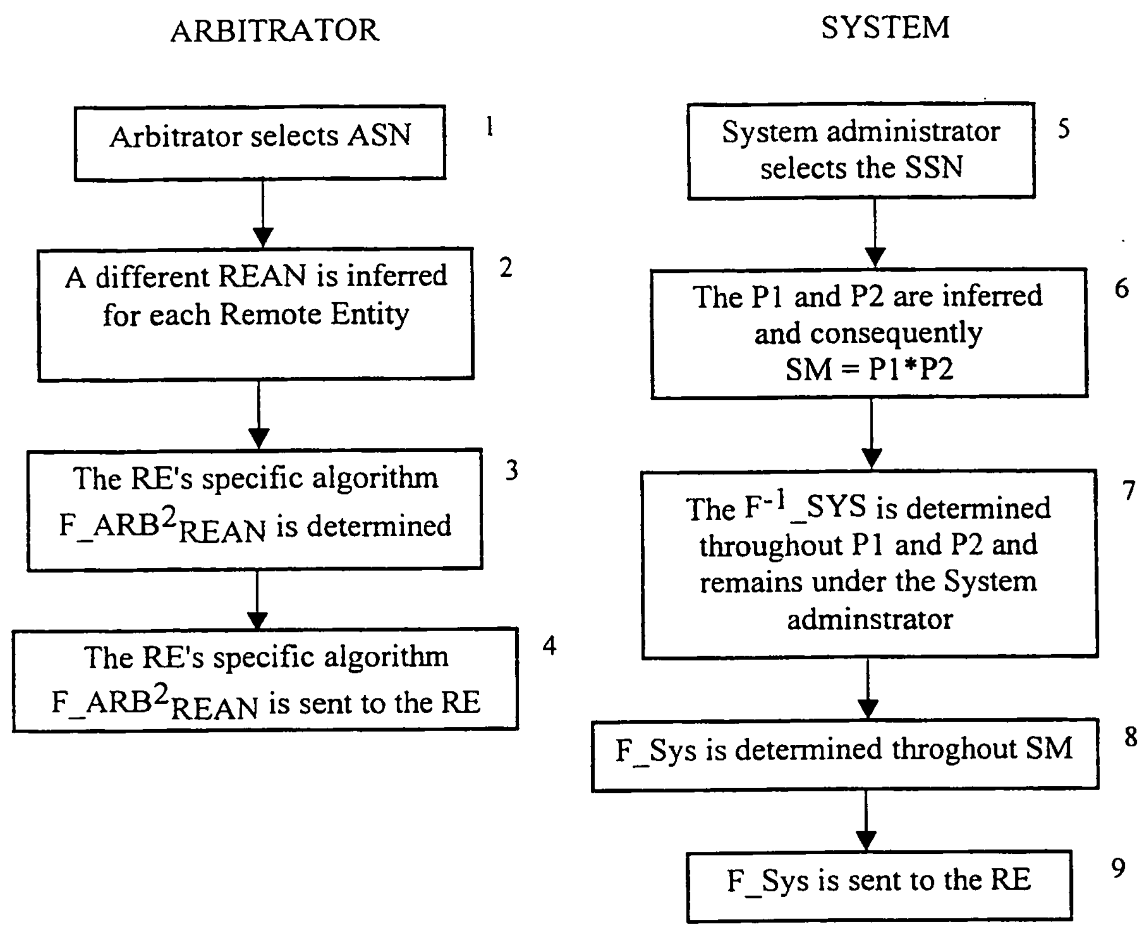

ARBITRATOR
SYSTEM
Arbitrator selects ASN
1
A different REAN is inferred for each Remote Entity
2
The RE's specific algorithm F_ARB^{2}REAN is determined
3
The RE's specific algorithm F_ARB^{2}REAN is sent to the RE
4
System administrator selects the SSN
5
The P1 and P2 are inferred and consequently SM = P1*P2
6
The F^{-1}_SYS is determined throughout P1 and P2 and remains under the System adminstrator
7
F_Sys is determined throghout SM
8
F_Sys is sent to the RE
9

FIGURE 2

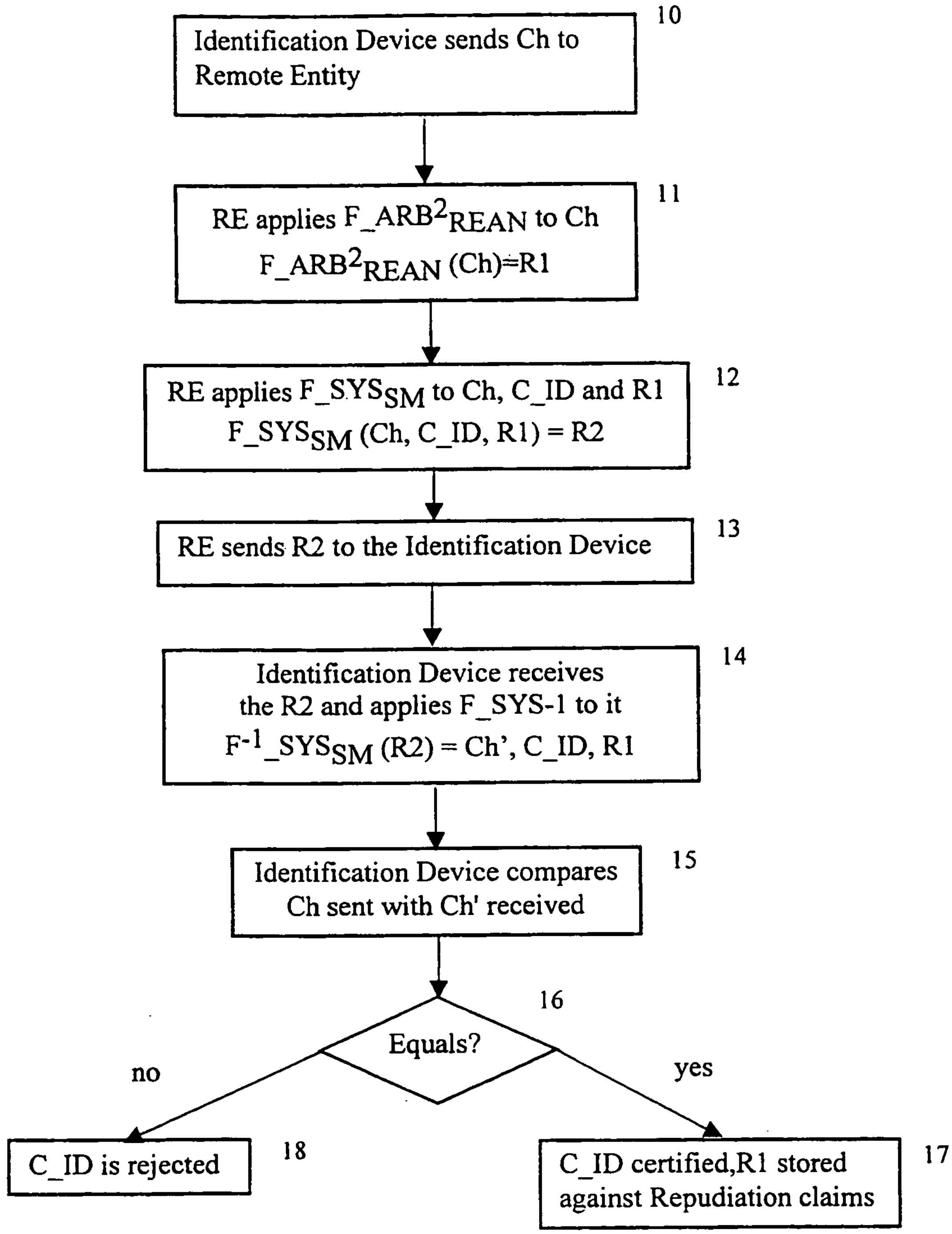

10
Identification Device sends Ch to Remote Entity
11
RE applies F_ARB^2$_{REAN}$ to Ch
F_ARB^2$_{REAN}$ (Ch)=R1
12
RE applies F_SYS$_{SM}$ to Ch, C_ID and R1
F_SYS$_{SM}$ (Ch, C_ID, R1) = R2
13
RE sends R2 to the Identification Device
14
Identification Device receives the R2 and applies F_SYS-1 to it
F^{-1}_SYS$_{SM}$ (R2) = Ch', C_ID, R1
15
Identification Device compares Ch sent with Ch' received
16
Equals?
no
yes
C_ID is rejected
18
C_ID certified,R1 stored against Repudiation claims
17

FIGURE 3

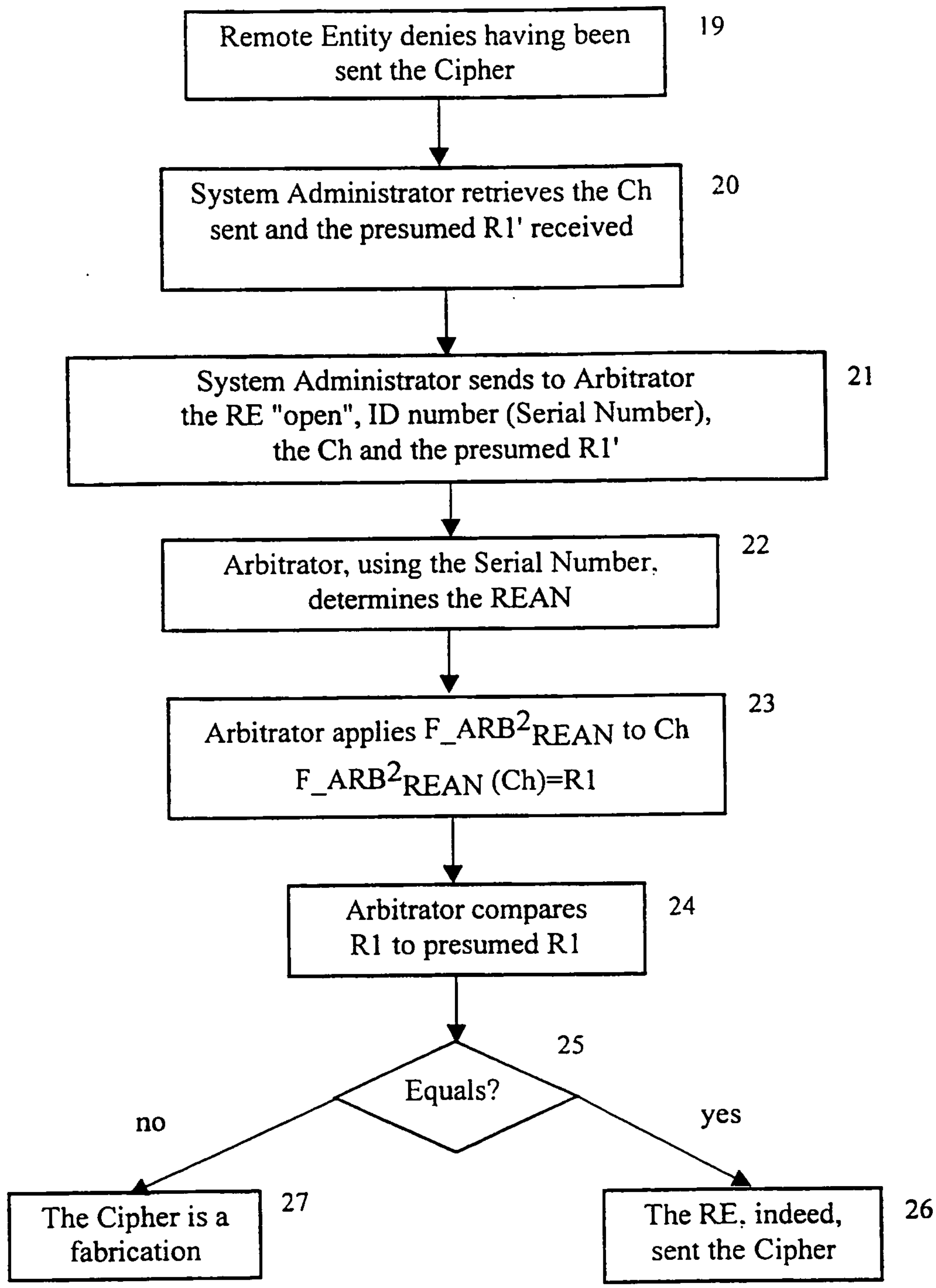

Remote Entity denies having been sent the Cipher
19
System Administrator retrieves the Ch sent and the presumed R1' received
20
System Administrator sends to Arbitrator the RE "open", ID number (Serial Number), the Ch and the presumed R1'
21
Arbitrator, using the Serial Number, determines the REAN
22
Arbitrator applies $F\_ARB^2_{REAN}$ to Ch
$F\_ARB^2_{REAN}(Ch)=R1$
23
Arbitrator compares R1 to presumed R1
24
Equals?
25
no
yes
The Cipher is a fabrication
27
The RE, indeed, sent the Cipher
26

FIGURE 4

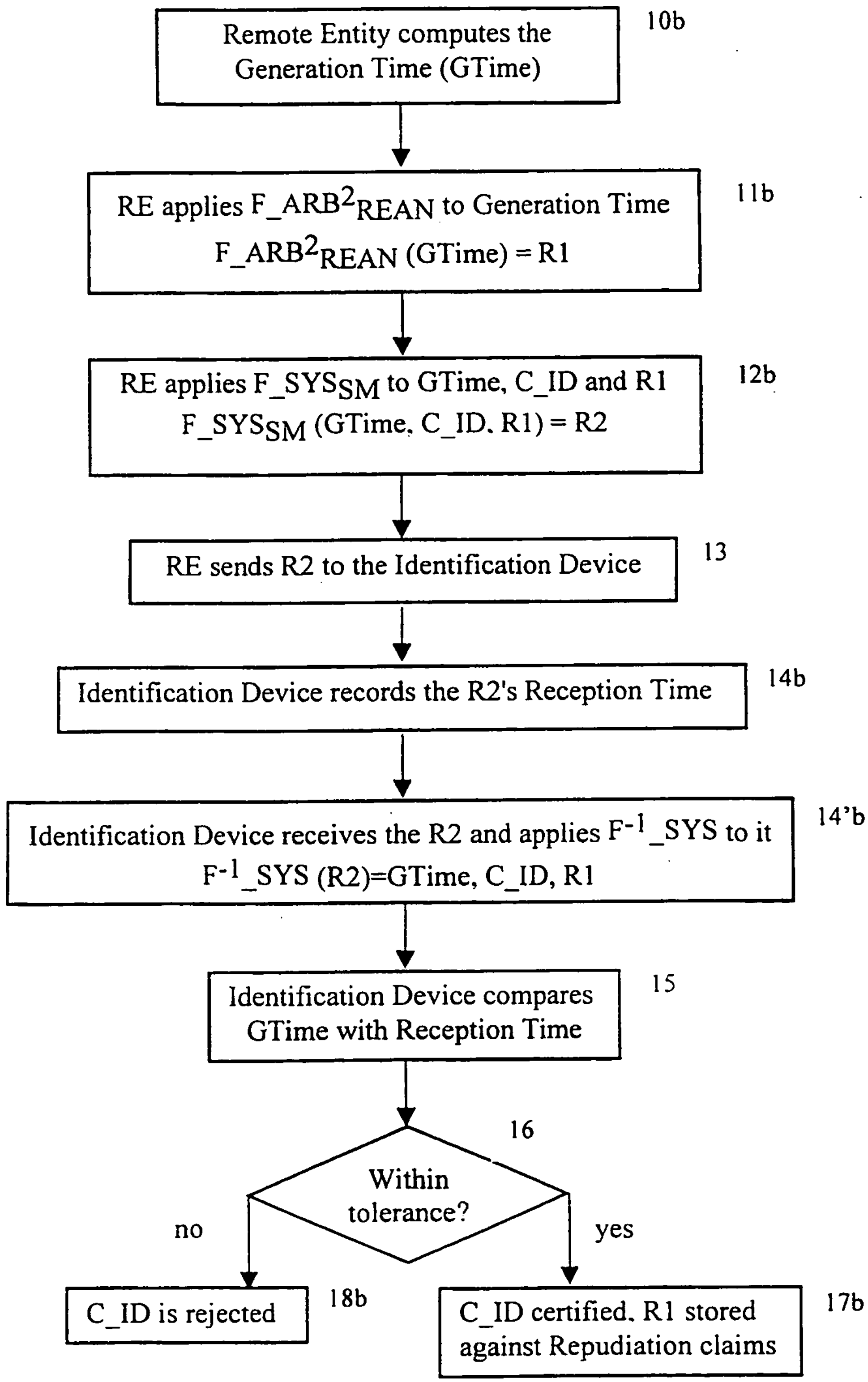

Remote Entity computes the Generation Time (GTime)
10b
RE applies F_ARB2REAN to Generation Time
F_ARB2REAN (GTime) = R1
11b
RE applies F_SYSSM to GTime, C_ID and R1
F_SYSSM (GTime, C_ID, R1) = R2
12b
RE sends R2 to the Identification Device
13
Identification Device records the R2's Reception Time
14b
Identification Device receives the R2 and applies F-1_SYS to it
F-1_SYS (R2)=GTime, C_ID, R1
14'b
Identification Device compares GTime with Reception Time
15
Within tolerance?
16
no
yes
C_ID is rejected
18b
C_ID certified, R1 stored against Repudiation claims
17b

FIGURE 5

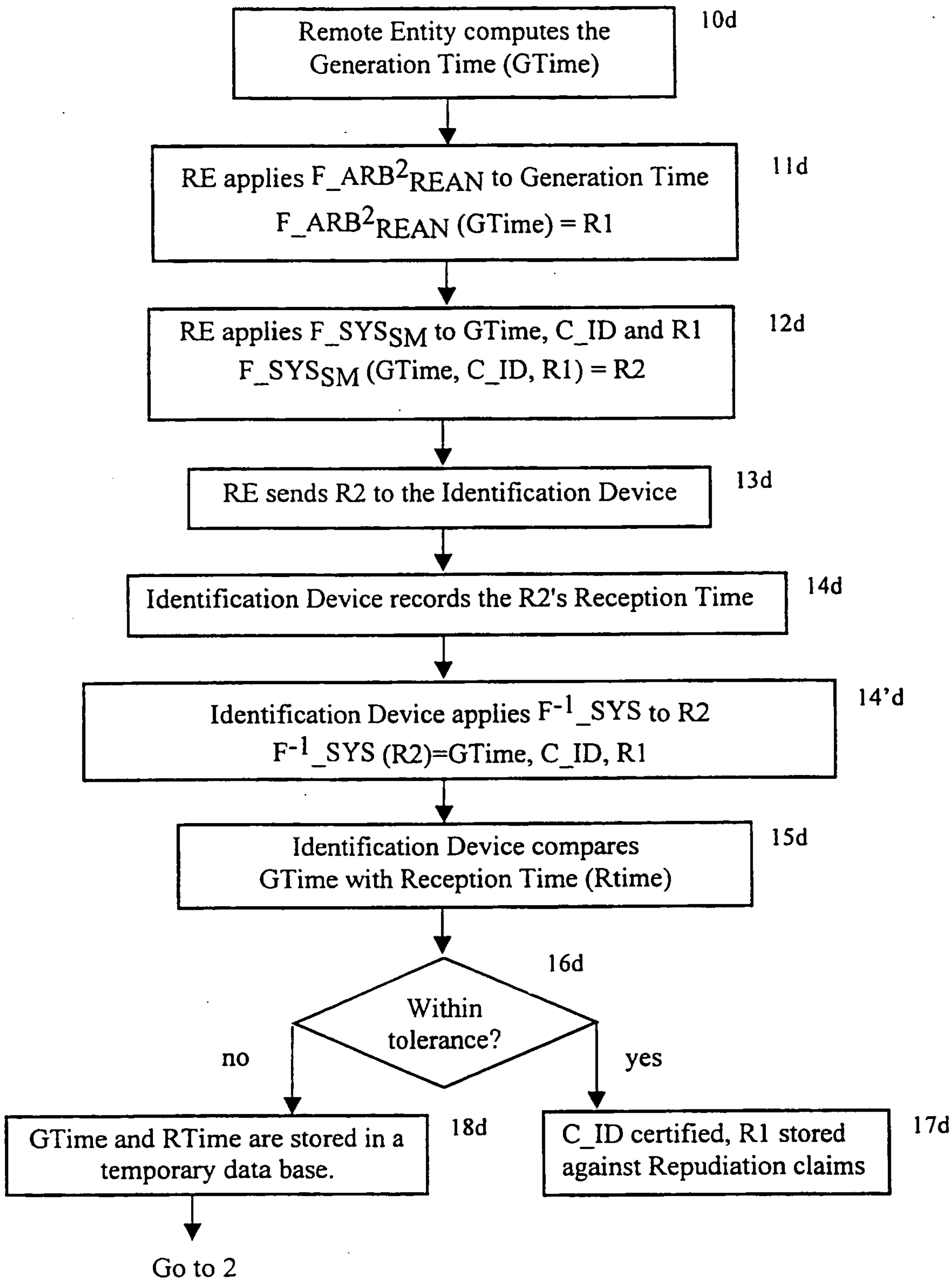

Remote Entity computes the Generation Time (GTime)
10d
RE applies $F\_ARB^2{}_{REAN}$ to Generation Time
$F\_ARB^2{}_{REAN}$ (GTime) = R1
11d
RE applies F_SYS_{SM} to GTime, C_ID and R1
F_SYS_{SM} (GTime, C_ID, R1) = R2
12d
RE sends R2 to the Identification Device
13d
Identification Device records the R2's Reception Time
14d
Identification Device applies F^{-1}_SYS to R2
F^{-1}_SYS (R2)=GTime, C_ID, R1
14'd
Identification Device compares GTime with Reception Time (Rtime)
15d
Within tolerance?
16d
no
yes
GTime and RTime are stored in a temporary data base.
18d
C_ID certified, R1 stored against Repudiation claims
17d
Go to 2

FIGURE 5 (continued)

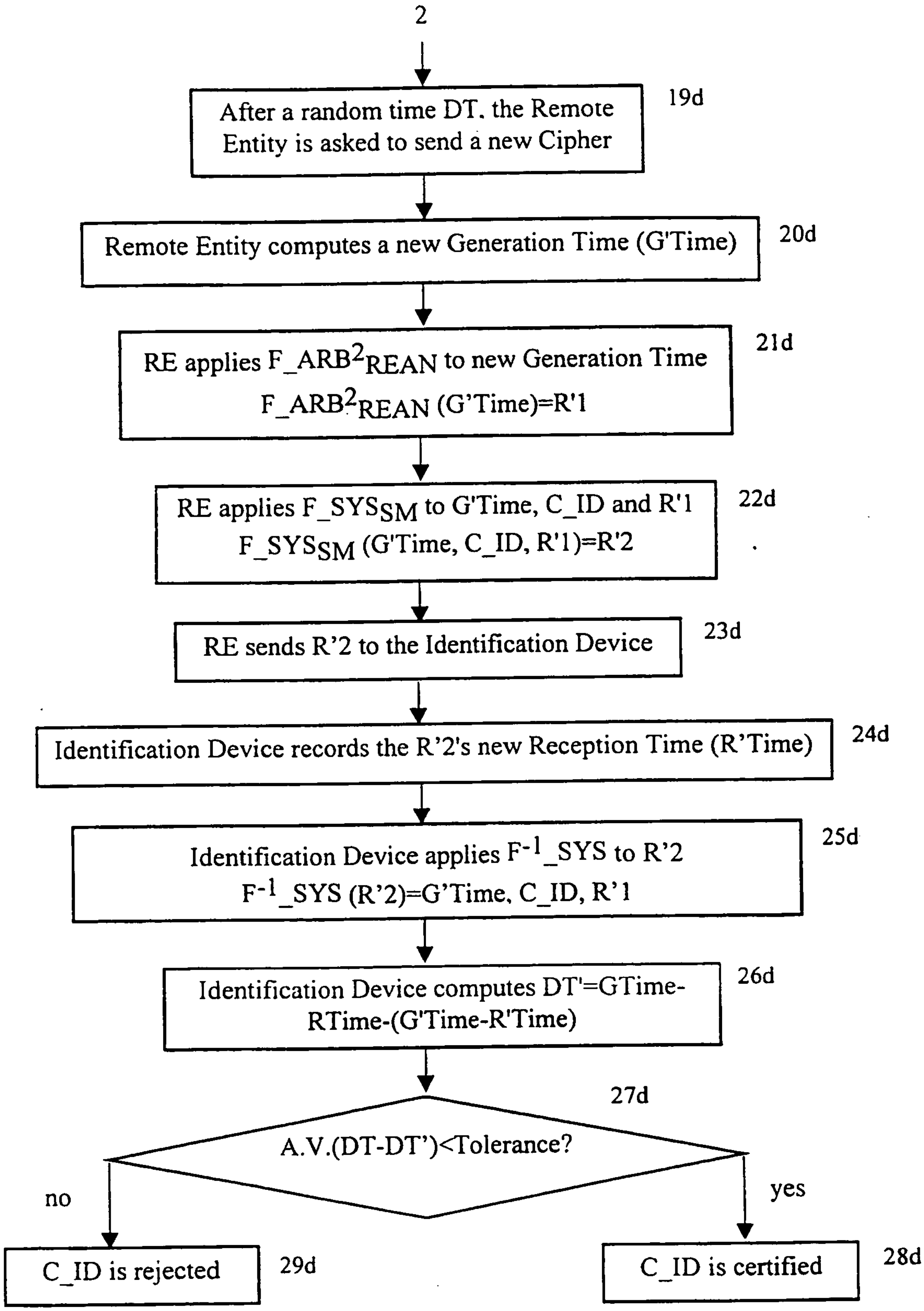


FIGURE 6

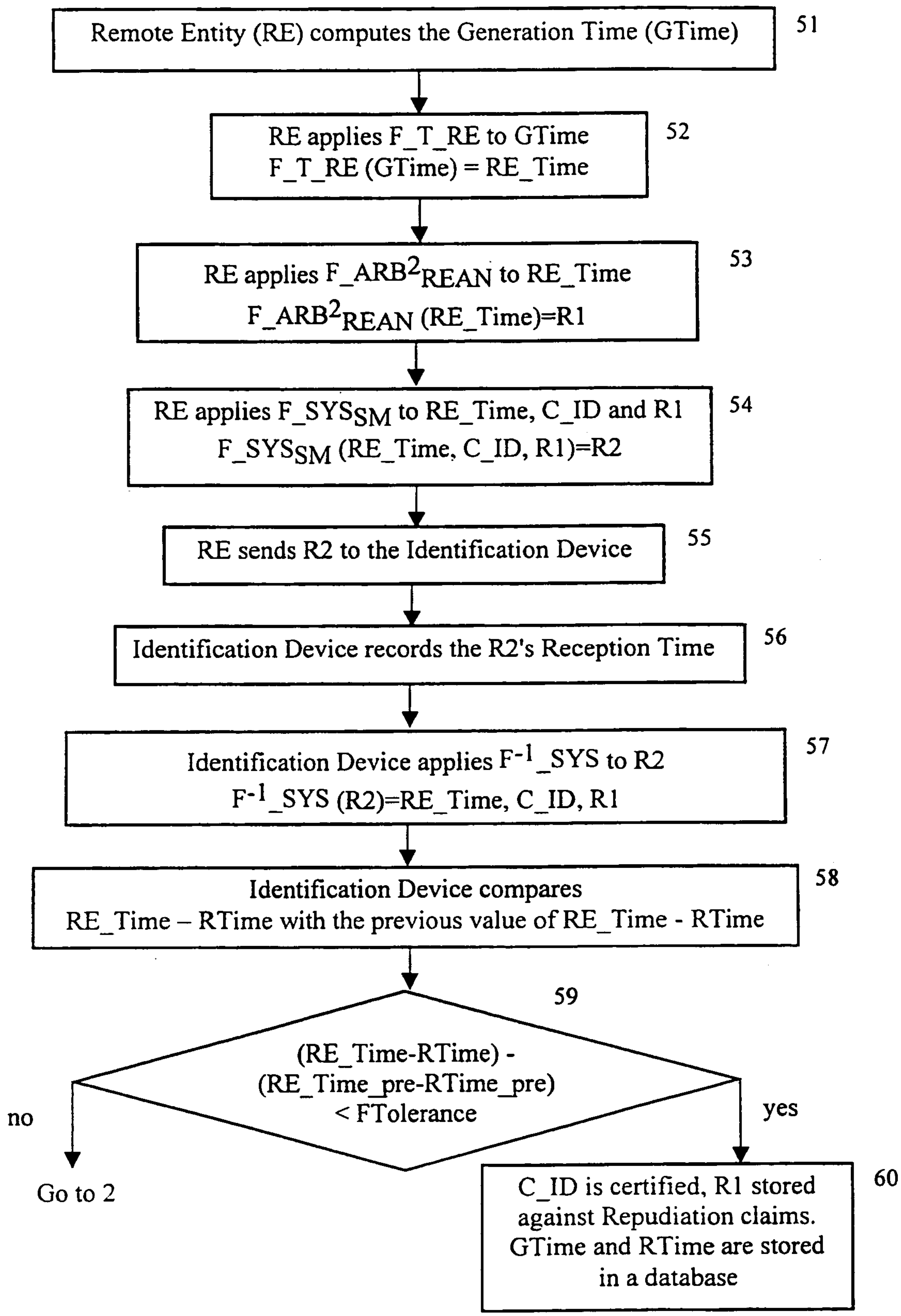

Remote Entity (RE) computes the Generation Time (GTime)
51
RE applies F_T_RE to GTime
F_T_RE (GTime) = RE_Time
52
RE applies F_ARB^{2}REAN to RE_Time
F_ARB^{2}REAN (RE_Time)=R1
53
RE applies F_SYS$_{SM}$ to RE_Time, C_ID and R1
F_SYS$_{SM}$ (RE_Time, C_ID, R1)=R2
54
RE sends R2 to the Identification Device
55
Identification Device records the R2's Reception Time
56
Identification Device applies F^{-1}_SYS to R2
F^{-1}_SYS (R2)=RE_Time, C_ID, R1
57
Identification Device compares
RE_Time – RTime with the previous value of RE_Time - RTime
58
59
(RE_Time-RTime) -
(RE_Time_pre-RTime_pre)
< FTolerance
no
yes
Go to 2
C_ID is certified, R1 stored
against Repudiation claims.
GTime and RTime are stored
in a database
60

FIGURE 6 (continued)

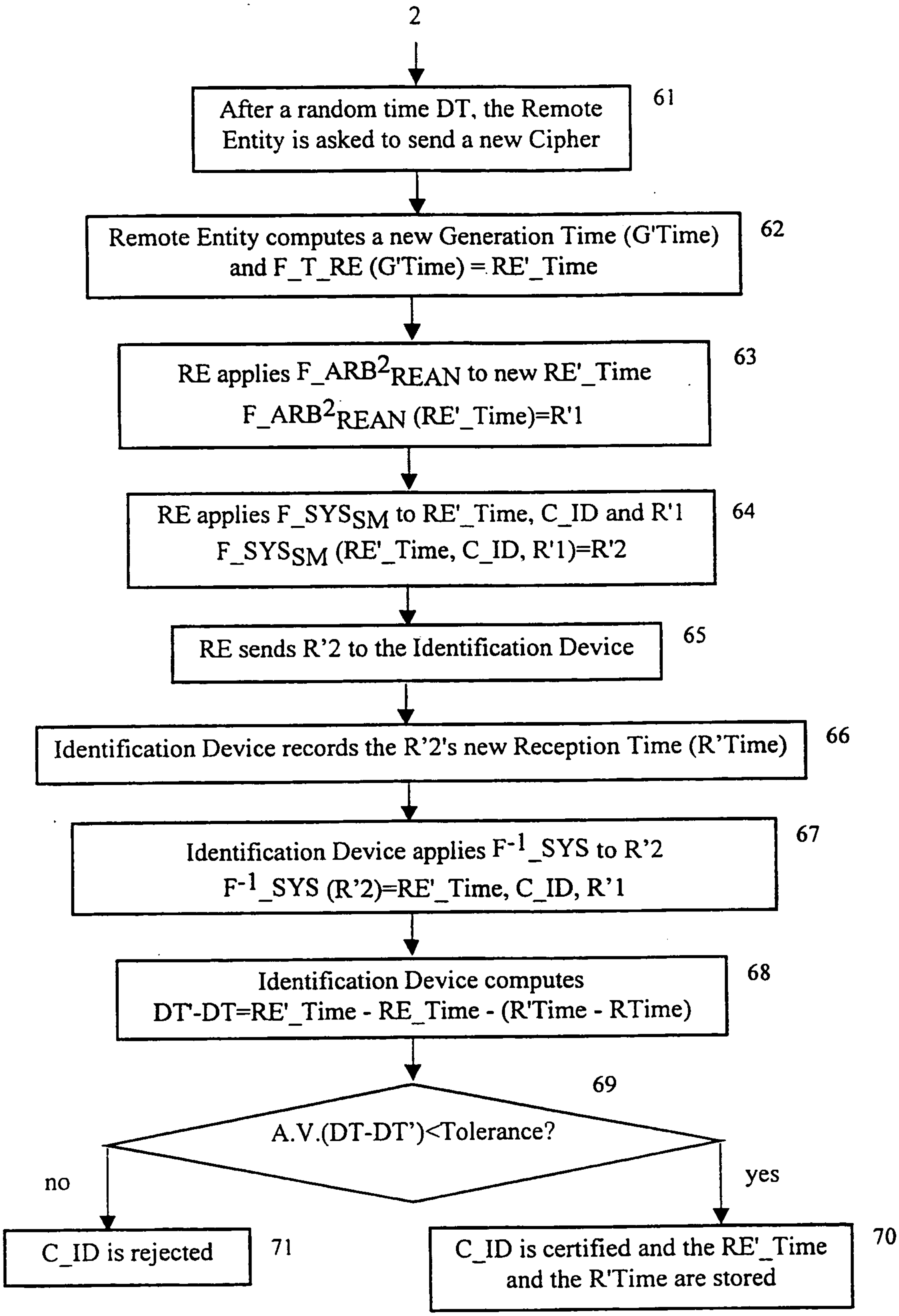
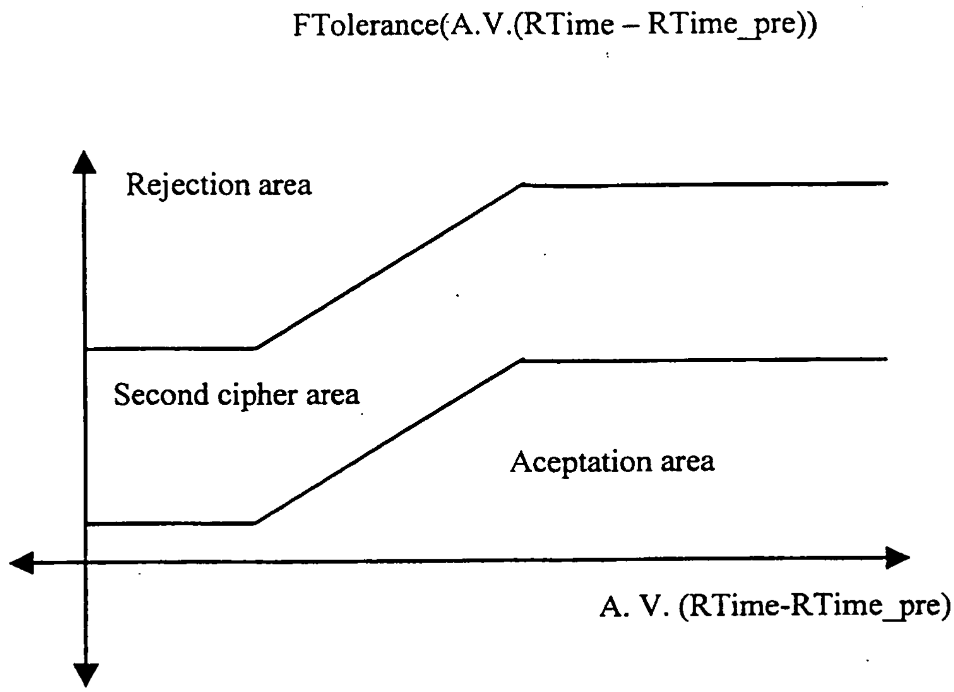


FIGURE 7


FTolerance(A.V.(RTime – RTime_pre))
Rejection area
Second cipher area
Aceptation area
A. V. (RTime-RTime_pre)